

# **SISTEM PENGENALAN INDIVIDU MELALUI IDENTIFIKASI TELAPAK TANGAN DENGAN MENGGUNAKAN MATRIKS DISKRIMINATOR**

**Nama Mahasiswa : Ahmad Sirojuddin Luthfi**  
**NRP : 1210 100 052**  
**Jurusan : Matematika**  
**Dosen Pembimbing : Drs. Nurul Hidayat, M.Kom**

## **Abstrak**

Sistem biometrika merupakan teknologi pengenalan individu dengan menggunakan bagian tubuh atau perilaku manusia. Sidik jari dan telapak tangan, masing-masing contoh biometrika berdasarkan bagian tubuh dan tingkah laku manusia. Sistem pengenalan diri adalah sistem untuk mengenali identitas seseorang secara otomatis dengan menggunakan teknologi komputer. Sistem akan mencari dan mencocokkan identitas seseorang dengan suatu basis data acuan yang telah disiapkan sebelumnya melalui proses pendaftaran. Permukaan area telapak tangan yang lebih luas dibandingkan sidik jari diharapkan dapat menghasilkan ciri yang memiliki kemampuan membedakan yang lebih handal. Telapak tangan memiliki beberapa karakteristik unik diantaranya ciri-ciri geometri seperti panjang, lebar, dan area telapak tangan, garis-garis prinsip seperti garis hati, garis kepala, garis kehidupan, garis-garis kusut dan ciri-ciri minusi. Dari ciri-ciri yang ada itu akan dikembangkan suatu sistem pengenalan individu melalui identifikasi dengan telapak tangan dengan matriks diskriminator. Diskriminator adalah suatu pembeda antara telapak tangan yang satu dengan telapak tangan yang lain. Diskriminator ini didapat dari proses ekstraksi ciri.

**Kata kunci : Biometrika, Diskriminator, Ekstraksi Ciri, Telapak Tangan**

## **INDIVIDUAL RECOGNITION SYSTEM BY PALMPRINT IDENTIFICATION USING MATRIX DISCRIMINATOR**

**Name** : *Ahmad Sirojuddin Luthfi*  
**NRP** : *1210 100 052*  
**Department** : *Mathematic*  
**Supervisor** : *Drs. Nurul Hidayat, M.Kom*

### ***Abstract***

*Biometric systems is the individual recognition technologies using body parts or human behavior. Fingerprints and palmprint, each instance based on the body's biometrics and human behavior. Individual recognition system is a system for automatically recognizing a person's identity by using computer technology. The system will search for and match a person's identity with a reference database that has been prepared previously through the registration process. The surface palm print wider than expected to generate a fingerprint characteristic that has the ability to distinguish more reliable. Palmprint has some unique characteristics A geometry characteristics such as length, width, and area of the palm, the lines of principles such as the heart line, the head line, life line, lines tangled and minusi characteristics. Of the existing characteristics that will be developed a system of individual recognition through identification with the palm of the hand with the matrix discriminator. Discriminator is a difference between the palm of one hand with the other hand. Discriminator is obtained from the feature extraction process.*

**Key words** : *Biometric, Discriminator, Feature Extraction, Palmprint*

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Studi Penelitian Sebelumnya**

Pada tahun 2012, artikel yang ditulis J. Guo dengan judul “*Palmprint Recognition Using Block Entropy Map From a Single Image Per Person*” telah berhasil mencocokkan telapak tangan dengan akurasi 77.6% [2]. Pada artikel yang ditulis Guo juga membandingkan *Entropy Map* (EM) dengan metode *statistics feature*, *Fourier Transform* dan *DCT Transform*. Metode EM menggunakan segmentasi blok dan membagi citra telapak tangan menjadi beberapa *sub-image*. Dengan menjadikan citra telapak tangan menjadi *sub-image*, berarti EM memproses detail dari citra telapak tangan. Jika terjadi perubahan atau ada *noise* pada citra telapak tangan, maka EM kurang optimal. Pada tahun 2012, Tugas Akhir Jurusan Matematika ITS yang ditulis oleh Ricky Kurniadi dengan judul “Pengembangan Sistem Identifikasi Telapak Tangan dengan Menggunakan Metode Filter Bank Gabor” dengan hasil pengujian identifikasi telapak tangan pengguna, diperoleh unjuk kerja yang cukup tinggi, yang ditunjukkan oleh persentase tingkat keberhasilan sistem dalam melakukan identifikasi pengguna sebesar 92,727% dan dengan metode ini dapat memproses citra telapak tangan yang mengalami perubahan ataupun ada *noise* [3].

#### **2.2 Teori Penunjang**

##### **2.2.1 Biometrika**

Secara harfiah, biometrika atau biometrics berasal dari kata bio dan metrics. Bio berarti sesuatu yang hidup, dan metrics berarti mengukur. Biometrika berarti mengukur karakteristik pembeda (*distinguishing traits*) pada badan atau perilaku seseorang yang digunakan untuk melakukan pengenalan secara otomatis terhadap identitas orang tersebut, dengan membandingkannya dengan karakteristik yang sebelumnya telah disimpan pada suatu database [3].

Secara umum karakteristik pembeda tersebut dapat dikelompokkan menjadi 2, yaitu karakteristik fisiologis atau fisik (*physiological/physical characteristic*) dan karakteristik perilaku (*behavioral characteristic*). Biometrika berdasarkan karakteristik fisiologis/fisik menggunakan bagian-bagian fisik dari tubuh seseorang sebagai kode unik untuk pengenalan, seperti DNA, telinga, jejak panas pada wajah, geometri tangan, pembuluh tangan, sidik jari, iris mata, telapak tangan, retina, gigi dan bau (komposisi kimia) dari keringat tubuh. Sedangkan biometrik berdasarkan karakteristik perilaku menggunakan perilaku seseorang sebagai kode unik untuk melakukan pengenalan, seperti gaya berjalan, hentakan tombol, tanda tangan dan suara. Khusus untuk suara, lebih tepat disebut sebagai karakteristik gabungan, karena suara dibentuk berdasarkan karakteristik fisik (bagian-bagian fisik tubuh manusia yang memproduksi suara) dan karakteristik perilaku (cara atau logat seseorang dalam berbicara). Tidak semua bagian tubuh atau perilaku seseorang dapat digunakan sebagai biometrika. Ada beberapa persyaratan yang harus dipenuhi agar bagian-bagian tubuh atau perilaku manusia dapat digunakan sebagai biometrika, antara lain : Universal (*universality*), Membedakan (*distinctiveness*), Permanen (*permanence*), Kolektabilitas (*collectability*), Unjuk kerja (*performance*), Dapat diterima (*acceptability*), dan Tidak mudah dikelabui (*circumvention*).

## **2.2.2 Unjuk Kerja Sistem Biometrika**

Unjuk kerja suatu sistem biometrika sering dinyatakan dengan rasio kesalahan pencocokan (*matching error rate*), yaitu rasio kesalahan kecocokan (*false match rate*) dan rasio kesalahan ketidakcocokan (*false non match rate*).

### **2.2.2.1 Rasio Kesalahan Kecocokan**

*False Match Rate* (FMR) menyatakan probabilitas sampel dari pengguna cocok dengan acuan yang diambil secara acak

milik pengguna yang berbeda. *False match rate* disebut juga *false positive*. Rasio kesalahan kecocokan dihitung dengan rumus:

$$\text{Rasio Kesalahan Kecocokan} = \frac{\alpha}{n} \times 100\% \quad (2.1)$$

dengan :

$\alpha$  : jumlah kesalahan kecocokan

$n$  : jumlah keseluruhan proses pencocokan

#### 2.2.2.2 Rasio Kesalahan Ketidakkcocokan

*False Non Match Rate* (FNMR) menyatakan probabilitas sampel dari pengguna tidak cocok dengan acuan lain yang diberikan pengguna yang sama. *False non match rate* ini juga *false negative*. Rasio kesalahan ketidakkcocokan dihitung dengan rumus:

$$\text{Rasio Kesalahan Ketidakkcocokan} = \frac{\beta}{n} \times 100\%$$

dengan :

$\beta$  : jumlah kesalahan ketidakkcocokan

$n$  : jumlah keseluruhan proses pencocokan

#### 2.2.3 Pola Telapak Tangan

Telapak tangan mempunyai ciri lebih banyak daripada sidik jari. Ciri yang dimiliki telapak tangan adalah sebagai berikut:

a. Ciri geometri (*geometry feature*)

Ciri ini menyangkut bentuk geometri telapak tangan. Ciri geometri setiap individu hampir sama dan dapat dipalsukan dengan hanya membuat model tiruan suatu tangan. Ciri ini tidak dimungkinkan untuk digunakan pada sistem verifikasi, apalagi untuk sistem identifikasi, karena jumlah perbedaan antar individu sedikit sehingga kemampuan membedakannya rendah.

b. Ciri garis-garis utama (*principal-line feature*)

Garis-garis utama atau prinsip, dapat digunakan untuk membedakan antara satu orang dengan orang lain karena garis ini unik, stabil, dan sedikit mengalami perubahan dalam

kurun waktu yang cukup lama. Ada tiga jenis garis utama, yaitu: garis hati (*heart line*), garis kepala (*head line*), dan garis kehidupan (*life line*). Untuk ukuran basis data acuan yang cukup besar, sangat sulit untuk mendapatkan tingkat pengenalan tinggi dengan hanya menggunakan garis-garis ini karena kemiripannya dengan garis-garis utama telapak tangan orang yang berbeda.

- c. Ciri garis-garis kusut (*wrinkles feature*)  
Telapak tangan banya mengandung garis kusut atau tipis yang sifatnya berbeda dengan garis utama. Garis-garis ini lebih tipis dan tidak beraturan. Garis-garis ini mampu menghasilkan ciri yang lebih rinci.
- d. Ciri titik delta (*delta-point feature*)  
Ada 5 daerah delta, seperti daerah pada akar jari-jari dan diluar daerah jari-jari. Titik ini unik dan bersifat stabil, namun sulit untuk memperoleh ciri ini dari citra telapak tangan beresolusi rendah.
- e. Ciri minusi (*minutiae feature*)  
Minusi merupakan pola bukit dan lembah pada permukaan telapak tangan seperti pada sidik jari. Ciri minusi hanya dapat diperoleh pada citra telapak tangan beresolusi tinggi dan membutuhkan komputasi yang tinggi.

Ciri garis-garis utama dan garis-garis kusut sering disebut ciri garis saja. Ciri ini dapat diperoleh dari citra telapak tangan beresolusi rendah, ini merupakan kelebihan telapak tangan dibanding sidik jari [5].

#### 2.2.4 Sistem Identifikasi Telapak Tangan

Sistem identifikasi telapak tangan adalah sebuah sistem pengenalan pola yang digunakan untuk menspesifikasikan identitas seseorang berdasarkan telapak tangannya. Dengan tanpa diketahui identitas orang tersebut sebelumnya, sistem identifikasi telapak tangan berusaha untuk mencocokkan telapak tangan orang tersebut dengan keseluruhan data telapak tangan yang tersimpan di *database*. Sistem identifikasi telapak tangan terdiri dari tiga

tahapan dasar, yaitu Akuisisi telapak tangan (*palmprint acquisition*), Ekstraksi ciri (*feature extraction*), Pencocokan telapak tangan (*palmprint matching*) [3].

### 2.2.5 PCA (Principal Component Analysis)

PCA adalah suatu teknik untuk memproses suatu data yang berdimensi besar dan diproyeksikan ke suatu ruang yang berdimensi kecil [9]. Citra inputan ditentukan dengan resolusi  $32 \times 32$ , karena itu suatu vektor kolom/baris dapat diperoleh, dengan menyatakan sebuah titik dalam ruang berdimensi 1024.

Diberikan sampel dari citra telapak tangan yaitu  $x_1, x_2, \dots, x_M$ , dimana  $M$  adalah banyaknya citra dalam training set. Rata-rata citra ( $\mu$ ) dari training set didefinisikan dengan

$$\mu = \frac{1}{M} \sum_{i=1}^M x_i \quad (2.3)$$

Sehingga diperoleh matriks kovarian ( $C$ ) dari  $\{x_i\}$  sebagai berikut:

$$C = \frac{1}{M} \sum_{i=1}^M (x_i - \mu)(x_i - \mu)^T \quad (2.4)$$

Sehingga, matriks  $C$  berdimensi  $1.024 \times 1.024$ . Dari matriks  $C$  dapat ditentukan nilai eigen dan vektor eigennya [4]. Dengan PCA, vektor eigen ini diurutkan berdasarkan nilai eigennya yang bersesuaian dan diambil sebagian secara berurutan dari nilai eigen yang terbesar.

### 2.2.6 Matriks Diskriminator

Diskriminator memiliki banyak definisi, tapi dalam hal ini definisi yang paling tepat dari bahasa inggris, yang berasal dari kata *discriminator*. *Discriminator* menurut *Oxford*

*Dictionaries* adalah “a characteristic which enables people or things to be distinguished from one another”, yang artinya ciri khas yang memungkinkan orang atau sesuatu bisa dibedakan satu dengan yang lain. Matriks diskriminator adalah matriks yang digunakan untuk membedakan telapak tangan setiap individu.

Proses penghitungan diskriminator adalah sebagai berikut. Misalkan :

$$V = [v_1, v_2, v_3, \dots, v_n] = \begin{bmatrix} V_{11} & \cdot & V_{1n} \\ \cdot & \cdot & \cdot \\ V_{n1} & \cdot & V_{nn} \end{bmatrix} \quad (2.5)$$

dengan:

$[v_1, v_2, \dots, v_n]$  = eigen vektor dari matriks covarian yang sudah direduksi dari proses PCA.

$n$  = adalah panjang diskriminator.

Maka diskriminatornya adalah

$$D_i = V^T (x_i - \mu) \quad (2.6)$$

dengan:

$D_i$  = diskriminator untuk tiap citra telapak tangan

$V^T$  = matriks transpose dari eigen vektor yang sudah direduksi dengan PCA

$x_i$  = citra telapak tangan yang sudah di bentuk vektor kolom

$\mu$  = rata-rata dari semua citra di training set

## 2.2.7 Jarak Euclidian Ternormalisasi

Pengukuran kesamaan dihitung dengan persamaan jarak Euclidean ternormalisasi sebagai berikut [3]:



$$\bar{d}(u, v) = \left( \sum_{i=1}^n (\bar{u}_i - \bar{v}_i)^2 \right)^{\frac{1}{2}} \quad (2.7)$$

dengan:

$$\bar{u}_i = \frac{u_i}{\|u\|} \quad (2.8)$$

dan

$$\bar{v}_i = \frac{v_i}{\|v\|} \quad (2.9)$$

$\|u\|$  dan  $\|v\|$  masing-masing disebut dengan norm dari  $u$  dan norm dari  $v$  untuk jarak euclidean dan diperoleh dari persamaan sebagai berikut:

$$\|u\| = \left( \sum_{i=1}^n (\bar{u}_i)^2 \right)^{\frac{1}{2}} \quad (2.10)$$

dan

$$\|v\| = \left( \sum_{i=1}^n (\bar{v}_i)^2 \right)^{\frac{1}{2}} \quad (2.11)$$

dengan:

$\bar{d}(u, v)$  = jarak Euclidian ternormalisasi

$u$  = diskriminator dari citra yang diuji

$v$  = diskriminator dari citra yang telah didaftarkan

$\|u\|$  = norm dari  $u$

$\|v\|$  = norm dari  $v$

$\bar{u}$  = diskriminator ternormalisasi dari citra yang diuji

$\bar{v}$  = diskriminator ternormalisasi dari citra yang telah didaftarkan

$n$  = panjang diskriminator

Sedangkan untuk mendapatkan skor hasil pencocokan telapak tangan maka digunakan metode pengukuran kesamaan (*similarity measure*). Rentang dari skor berada antara 0 sampai 1 yang didapat dengan persamaan sebagai berikut:

$$skor = 1 - \frac{d(u,v)}{2} \quad (2.12)$$

Skor adalah suatu nilai yang merepresentasikan tingkat kemiripan telapak tangan yang diuji dengan yang tersimpan di basis data. Semakin besar skornya semakin mirip citra yang dibandingkan.

## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Objek Penelitian**

Objek penelitian pada tugas akhir yang diusulkan ini adalah telapak tangan manusia, yang meliputi dua hal, yaitu:

1. Jenis material yang akan diteliti  
Jenis material yang akan diteliti adalah telapak tangan kiri manusia yang difoto melalui *webcam* yang terhubung ke sebuah PC atau Laptop
2. Aspek penelitian  
Aspek penelitian yang digunakan adalah mengidentifikasi individu melalui telapak tangannya dengan matriks diskriminator yang didapat dari proses ekstraksi ciri. Dan dalam tugas akhir yang diusulkan ini digunakan PCA (*Principal Component Analysis*) dalam proses ekstraksi cirinya.

#### **3.2 Peralatan**

Peralatan penelitian yang dibutuhkan untuk untuk menyelesaikan tugas akhir yang ini adalah:

1. Perangkat lunak utama yang digunakan untuk membuat program simulasi sistem identifikasi individu melalui telapak tangan adalah Visual C#.
2. Perangkat keras yang digunakan adalah *webcam* dan Compaq 420 dengan spesifikasi Intel® Core™ 2 Duo CPU T6570 @2,10 GHz, RAM 4 GB DDR3, dan VGA Mobile Intel® 4 Series Express Chipset Family.

#### **3.3 Tahap Penelitian**

Tahap penelitian dalam penyelesaian tugas akhir ini ada 4 tahap, yaitu:

### 1. Perancangan *Interface*

Pada tahap ini dirancang sebuah *Interface* yang *user friendly* sehingga dapat memudahkan pengguna dalam menggunakan sistem ini.

### 2. Perancangan dan implementasi sistem

Pada tahap ini akan dirancang sebuah sistem pengenalan individu melalui identifikasi telapak tangan dengan menggunakan matriks diskriminator. Gambaran umum dari sistem identifikasi telapak tangan yang akan dikembangkan adalah sebagai berikut:

- a. Akuisisi data: proses pengambilan citra telapak tangan dengan webcam.
- b. Pra-pemrosesan: proses pembuatan telapak tangan yang telah diakuisisi menjadi file citra telapak tangan ternormalisasi yang siap untuk diekstraksi cirinya.
- c. Ekstraksi ciri: proses untuk mendapatkan ciri-ciri citra telapak tangan yang dikenal sebagai diskriminator.
- d. Pendaftaran referensi: proses penyimpanan diskriminator ke dalam basis data acuan beserta dengan identitas pemiliknya.
- e. Pencocokan: proses pengukuran kesamaan antara citra input dengan citra referensi untuk menghasilkan suatu skor.

Setiap hasil dari perancangan sistem akan diimplementasikan menjadi sebuah program.

### 3. Integrasi program menjadi sebuah sistem

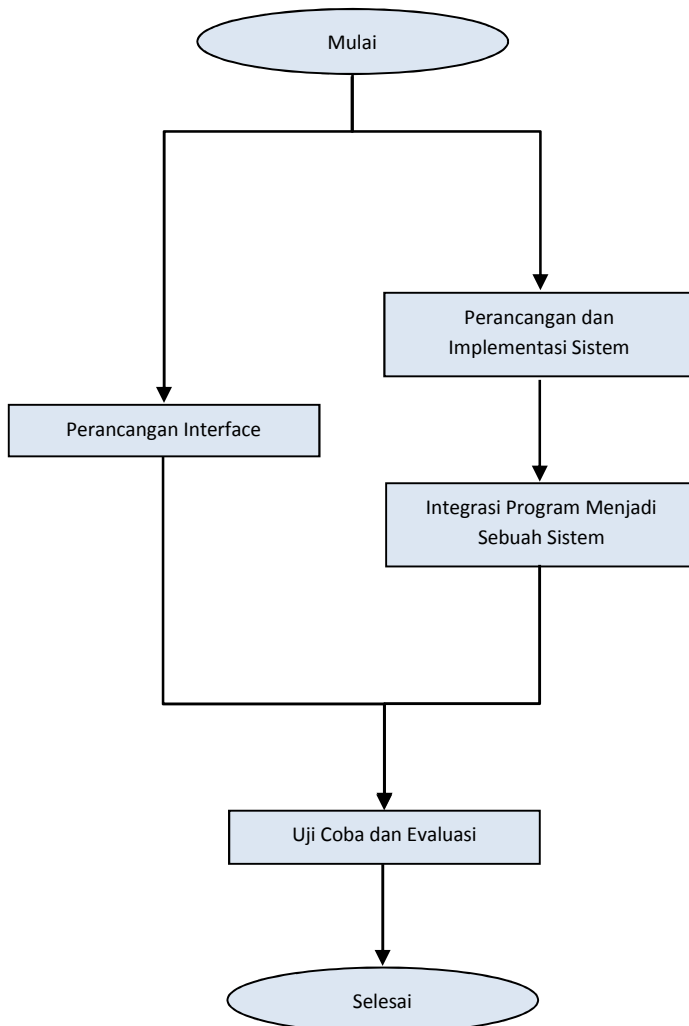
Pada tahap ini akan dilakukan integrasi untuk semua program yang telah diimplementasikan pada tahap sebelumnya sehingga dihasilkan sebuah sistem identifikasi telapak tangan yang terintegrasi.

### 4. Uji coba dan evaluasi

Pada tahap ini implementasi sistem akan diuji dengan menerapkan beberapa skenario percobaan untuk kemudian dianalisis dan dievaluasi hasilnya. Berikut ini adalah langkah-langkah untuk uji coba:

- a. Akuisisi citra telapak tangan, akuisisi merupakan proses pengambilan citra telapak tangan. Akuisisi telapak tangan dilakukan sebanyak 3 kali. Satu citra telapak tangan digunakan sebagai referensi, simpan pada folder Data Utama dan 2 citra lainnya digunakan sebagai uji coba, disimpan pada folder Data Uji Coba.
- b. Pendaftaran citra telapak tangan sebagai Referensi. Pendaftaran citra telapak tangan dilakukan setelah akuisisi. Pendaftaran ini menyimpan data pemilik citra telapak tangan ke sistem. Dan membentuk 4 buah diskriminator dengan ukuran yang berbeda utk setiap telapak tangannya.
- c. Kemudian megidentifikasi citra telapak tangan menggunakan 1 citra referensi dan 2 citra telapak tangan uji coba.
- d. Kemudian ketika semua citra telapak tangan sudah di uji, lakukan langkah c dengan mengganti ukuran diskriminatornya.

Diagram Alir dalam tahap penelitian dapat dilihat pada Gambar 3.1.



**Gambar 3.1: Diagram Alir Tahap Penelitian**

## **BAB IV**

### **PERANCANGAN DAN IMPLEMENTASI SISTEM**

Pada bab ini dijelaskan mengenai perancangan sistem dan hasil implementasi berbagai proses yang telah dirancang pada perancangan sistem. Pembahasan perancangan sistem diawali dengan penjelasan tentang lingkungan perancangan sistem, perancangan data, gambaran sistem identifikasi telapak tangan dengan menggunakan matriks diskriminator secara umum, dan perancangan algoritma proses-proses yang ada dalam tugas akhir ini. Selanjutnya membahas implementasi sistem yang dimulai dari lingkungan implementasi sistem dan dilanjutkan dengan hasil implementasi antarmuka dan keseluruhan proses di dalam sistem.

#### **4.1 Perancangan Sistem**

##### **4.1.1 Lingkungan Perancangan Sistem**

Lingkungan perancangan sistem yang akan dibangun meliputi beberapa perangkat keras dan perangkat lunak komputer. Detail spesifikasi lingkungan perancangan sistem dapat dilihat pada Tabel 4.1.

**Tabel 4.1 Spesifikasi Lingkungan Perancangan Sistem**

|                 |                |                                                                                             |
|-----------------|----------------|---------------------------------------------------------------------------------------------|
| Perangkat Keras | Prosesor       | : Intel® Core™ 2 Duo CPU T6570 @2,10 GHz                                                    |
|                 | Memory         | : 4 GB DDR3                                                                                 |
| Perangkat Lunak | Sistem Operasi | : Windows 7 Ultimate                                                                        |
|                 | Tools          | : Microsoft Visual C#<br>Aforge.NET ( <i>framework</i> )<br>Accord.NET ( <i>framework</i> ) |

##### **4.1.2 Perancangan Data**

Data yang digunakan dalam sistem ini dibagi menjadi 3 macam, yaitu data masukan, data proses, dan data keluaran. Data masukan merupakan data input dari pengguna sistem. Data proses adalah data ketika tahap-tahap pemrosesan telapak tangan sedang

dilakukan. Sedangkan data keluaran adalah data yang ditampilkan kepada pengguna sistem.

#### 4.1.2.1 Data Masukkan

Data masukan merupakan data yang dimasukkan oleh pengguna sistem berupa citra telapak tangan yang dimilikinya. Citra telapak tangan ini dimasukkan melalui *webcam* yang sudah terhubung dengan sistem. Di samping itu, yang termasuk data masukan juga berupa identitas yang dimiliki pengguna sistem. Identitas ini terdiri dari nomor id, nama, asal, dan foto wajah. Identitas ini digunakan untuk mendaftarkan pengguna ke dalam sistem dan sebagai informasi hasil identifikasi telapak tangan pengguna sistem.

#### 4.1.2.2 Data Proses

Data proses yang digunakan dalam sistem yang meliputi nama data, tipe data, dan keterangannya. Tipe data yang digunakan merujuk pada tipe-tipe data yang digunakan dalam bahasa pemrograman C#. Data proses selengkapnya disajikan dalam Tabel 4.2.

**Tabel 4.2 Data Proses**

| No | Nama Data            | Tipe Data | Keterangan                                                                          |
|----|----------------------|-----------|-------------------------------------------------------------------------------------|
| 1  | Citra Ternormalisasi | Int       | Data ini merupakan matriks hasil representasi citra telapak tangan yang akan diolah |
| 2  | Mean                 | Double    | Data ini merupakan rata-rata piksel dari semua citra yang telah didaftarkan         |
| 3  | Kovarian             | Double    | Data ini merupakan matriks covarian dari semua citra telapak tangan yang terdaftar  |



| No | Nama Data    | Tipe Data | Keterangan                                                             |
|----|--------------|-----------|------------------------------------------------------------------------|
| 4. | Citra Tajam  | Int       | Data ini merupakan hasil proses penajaman citra telapak tangan         |
| 5  | Eigen Value  | Double    | Data ini merupakan nilai eigen dari matriks kovarian                   |
| 6  | Eigen Vektor | Double    | Data ini merupakan vektor eigen yang bersesuaian dengan nilai eigennya |

Tipe data int merupakan tipe data primitif yang digunakan untuk menyimpan bilangan bulat dan memiliki jangkauan antara -2.147.486.648 sampai dengan 2.147.486.648. Tipe data double merupakan tipe data primitif yang digunakan untuk menyimpan bilangan pecahan dan memiliki jangkauan nilai antara  $-1,8 \times 10^{308}$  sampai dengan  $-5,0 \times 10^{-324}$  dan  $5,0 \times 10^{-324}$  sampai dengan  $1,8 \times 10^{308}$ .

#### 4.1.2.3 Data Keluaran

Data keluaran yang dihasilkan dari sistem ini adalah keputusan apakah sebuah telapak tangan dapat dikenali atau tidak. Keputusan tersebut berdasar pada skor yang dihasilkan dari perbandingan diskriminator dari data pengujian dengan referensi yang ada di dalam basis data. Jika telapak tangan dapat dikenali maka data keluaran berupa identitas pemilik telapak tangan yang dikenali. Sebaliknya jika telapak tangan tidak dapat dikenali maka data keluaran berupa informasi identitas yang kosong.

#### 4.1.3 Gambaran Proses Secara Umum

Gambaran proses secara umum merupakan gambaran keseluruhan proses yang dilakukan sistem serta algoritma-algoritma yang digunakan untuk mengerjakan masing-masing proses tersebut, yang nantinya akan diterapkan dalam pembuatan sistem. Dalam pembuatan sistem ini membutuhkan *library* Aforge.NET dan Accord.NET. Aforge.NET merupakan *framework* (*library*) dari bahasa pemrograman untuk

pengembangan suatu perangkat lunak yang dapat digunakan oleh programmer untuk membaca hardware pada komputer, mengolah citra digital, dll. Accord.NET merupakan *framework (library)* dari bahasa pemrograman untuk pengembangan suatu perangkat lunak yang membutuhkan fungsi matematika, statistik, dll.

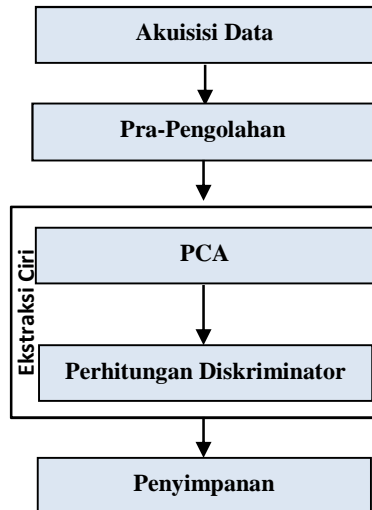
Sistem ini terdiri dari dua proses utama, yaitu:

1. Proses Registrasi / Pendaftaran

Dalam proses ini pengguna mendaftarkan identitas dirinya beserta data telapak tangan. Dalam proses ini terdapat beberapa proses yaitu:

- a. Proses akuisisi citra telapak tangan, proses ini bisa dilakukan secara langsung dengan menggunakan *webcam* maupun dengan membuka *file* citra telapak tangan yang sudah tersimpan sebelumnya.
- b. Proses pra-pengolahan, yaitu proses untuk mempersiapkan citra input untuk persiapan ekstraksi cirinya.
- c. Proses ekstraksi ciri, merupakan proses-proses yang dilakukan terhadap citra telapak tangan untuk memperoleh ciri citra telapak tangan dengan PCA dan telapak tangan ini akan direpresentasikan dalam sebuah matriks diskriminator.
- d. Proses penyimpanan telapak tangan, yang dibutuhkan sistem untuk menyimpan data dan ciri telapak tangan pengguna yang telah mendaftar.

Diagram Alir dari proses registrasi dapat dilihat pada Gambar 4.1.



**Gambar 4.1: Diagram Alir Proses Registrasi**

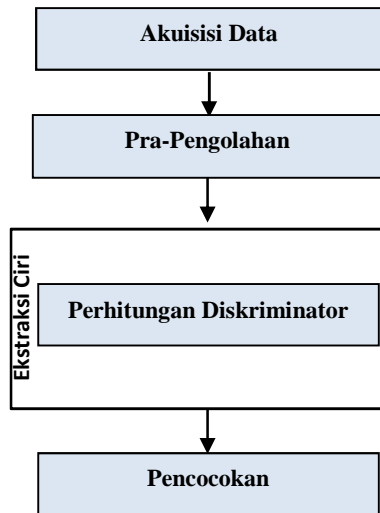
## 2. Proses Identifikasi / Pengenalan

Dalam proses ini pengguna asing, mengidentifikasi dirinya melalui citra telapak tangannya. Jika pengguna sudah terdaftar maka akan muncul identitasnya dan jika belum maka akan keluar identitas kosong. Dalam proses ini terdapat beberapa proses yaitu:

- Proses akuisisi citra telapak tangan, proses ini bisa dilakukan secara langsung dengan menggunakan *webcam* maupun dengan membuka *file* citra telapak tangan yang sudah tersimpan sebelumnya.
- Proses pra-pengolahan, yaitu proses untuk mempersiapkan citra input untuk persiapan ekstraksi cirinya.
- Proses ekstraksi ciri, dalam proses ekstraksi ini langsung merepresentasikan citra telapak tangan dengan matriks diskriminator.
- Proses pencocokan telapak tangan, proses penentuan skor dari diskriminator yang akan diidentifikasi dengan

diskriminator yang sudah tersimpan di basis data. Dari skor tersebut dicari yang terbesar dan jika melewati nilai ambang(*threshold*) maka pengguna teridentifikasi.

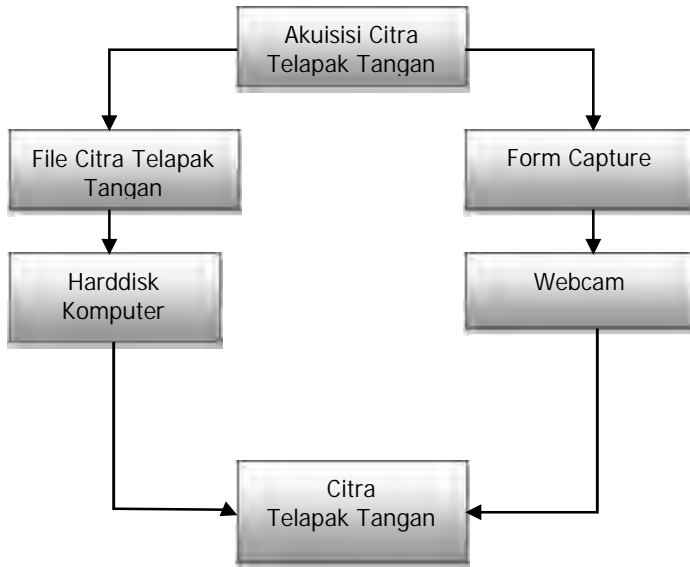
Diagram Alir dari proses identifikasi dapat dilihat pada Gambar 4.2.



**Gambar 4.2: Diagram Alir Proses Identifikasi**

#### **4.1.4 Proses Akuisisi Citra Telapak Tangan**

Akuisisi citra telapak tangan merupakan proses pengambilan citra telapak tangan pengguna sehingga dapat digunakan untuk melakukan beberapa proses yang lain terhadap citra telapak tangan tersebut. Pada tugas akhir ini, proses pengambilan citra telapak tangan dilakukan melalui dua metode. Metode pertama, citra telapak tangan diambil secara langsung dengan menggunakan *webcam* yang sudah terhubung ke sistem. Metode kedua, citra telapak tangan diambil secara tidak langsung yaitu dengan membuka citra telapak tangan yang sebelumnya sudah tersimpan di komputer. Proses yang dilakukan ketika proses akuisisi telapak tangan dapat dilihat pada Gambar 4.3.



**Gambar 4.3: Proses Akuisisi Citra Telapak Tangan**

Telapak tangan yang diakuisisi adalah telapak tangan sebelah kiri. Pengambilan telapak tangan didasarkan atas asumsi bahwa telapak tangan sebelah kiri tidak banyak mengalami perubahan, karena sedikit aktifitas.

#### **4.1.5 Proses Pra-pengolahan**

Pada proses akuisisi citra telapak tangan sebelumnya, citra yang diambil masih berupa suatu *file* yang belum bisa diekstraksi cirinya. Oleh karena itu harus dilakukan pra-pengolahan yang di dalamnya terdapat beberapa proses sebagai berikut:

##### **1. Konversi ke *Grayscale***

Citra telapak tangan yang akan diproses dalam tugas akhir ini adalah sebuah citra *grayscale*. Citra hasil akuisisi masih berjenis citra RGB, maka citra input yang sudah diubah

ukurannya dirubah menjadi citra *grayscale*. Prosesnya dilakukan dengan persamaan berikut :

$$grayscale = (0.2125 * R) + (0.7154 * G) + (0.0721 * B) \quad (4.1)$$

## 2. Normalisasi Ukuran

Citra yang didapat setelah proses akuisisi memiliki resolusi yang besar. Karena keterbatasan memory maka citra input yang akan digunakan dalam tugas akhir ini adalah 32 x 32. Di proses ini citra hasil akuisisi akan diubah ukurannya menjadi 32 x 32.

## 3. Normalisasi Intensitas

Setelah diubah ke citra *grayscale*, proses selanjutnya adalah melakukan normalisasi citra telapak tangan. Proses ini berguna untuk menyeragamkan nilai keabuan citra telapak tangan, baik yang termasuk alur maupun yang bukan termasuk alur. Rumus yang digunakan untuk melakukan proses normalisasi citra telapak tangan adalah sebagai berikut [3]:

$$N_i(x, y) = \begin{cases} M_0 + \sqrt{\frac{V_0\{(I(x, y) - M_i)^2\}}{V_i}} & \text{jika } I(x, y) > M_i \\ M_0 - \sqrt{\frac{V_0\{(I(x, y) - M_i)^2\}}{V_i}} & \text{jika } I(x, y) < M_i \end{cases} \quad (4.2)$$

dengan:

$N_i(x, y)$  = derajat nilai keabuan yang telah dinormalisasi pada piksel (x,y)

$I(x, y)$  = nilai keabuan pada piksel (x,y)

$M_i$  = estimasi nilai tengah citra asli

$M_0$  = nilai rata-rata yang ditetapkan/diinginkan

$V_i$  = nilai varian citra asli

$V_0$  = nilai varian yang ditetapkan/diinginkan

#### 4. Mepertajam citra

Sebelum diekstraksi cirinya citra telapak tangan ini perlu dipertajam dulu agar lebih terlihat ciri garisnya. Proses penajaman citra ini dilakukan dengan menggunakan matriks konvolusi berukuran  $3 \times 3$  yaitu

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Matriks konvolusi ini digerakkan mengelilingi citra telapak tangan yang sudah dinormalisasi tadi.

#### 5. Konversi matriks citra menjadi vektor kolom

Suatu citra telapak tangan direpresentasikan dengan sebuah matriks  $n \times n$ , namun untuk proses ekstraksi ciri dengan PCA matriks ini diubah bentuknya menjadi sebuah vektor kolom berukuran  $n^2 \times 1$ . Dengan bentuk inilah citra telapak tangan siap untuk diekstraksi cirinya.

### 4.1.6 Proses Ekstraksi Ciri

Ekstraksi ciri citra telapak tangan bertujuan untuk menentukan ciri-ciri dari suatu citra telapak tangan yang mampu membedakan antara telapak tangan yang satu dengan telapak tangan yang lain yang kemudian disebut diskriminator. Untuk mengubah suatu citra telapak tangan hingga menjadi suatu diskriminator, terdapat dua proses yang dilakukan, yaitu:

1. Proses PCA yaitu mencari vektor eigen dari matriks kovarian yang merupakan rata-rata dari citra yang didaftarkan.
2. Menghitung diskriminator untuk setiap citra telapak tangannya.

Berikut ini dijelaskan kedua proses yang digunakan pada proses ekstraksi ciri citra telapak tangan.

#### 4.1.6.1 Proses PCA

Diberikan 30 sampel dari citra telapak tangan yaitu  $x_1, x_2, \dots, x_{30}$  dimana. Rata-rata citra ( $\mu$ ) dari training set didefinisikan dengan

$$\mu = \frac{1}{30} \sum_{i=1}^{30} x_i \quad (4.3)$$

Sehingga diperoleh matriks kovarian ( $C$ ) sebagai berikut:

$$C = \frac{1}{30} \sum_{i=1}^{30} (x_i - \mu)(x_i - \mu)^T \quad (4.4)$$

Matriks  $C$  berdimensi  $1024 \times 1024$ , sehingga dapat ditentukan nilai eigen dan vektor eigennya [4]. Dengan PCA, vektor eigen ini diurutkan dari yang terbesar berdasarkan eigen valuenya yang bersesuaian dan reduksi menjadi sebanyak  $N = 50, 100, 150$ , dan  $200$  [4]. Kemudian dibentuk matriks dari vektor eigen yang sudah direduksi ini dan disimpan dalam suatu *text file*.

#### 4.1.6.2 Proses Perhitungan Diskriminator

Langkah terakhir pada proses ekstraksi ciri adalah proses penghitungan diskriminator. Dari proses PCA didapat 4 jenis matriks vektor eigen dengan ukuran yang berbeda-beda yaitu  $V_{1024 \times 50}$ ,  $V_{1024 \times 100}$ ,  $V_{1024 \times 150}$ , dan  $V_{1024 \times 200}$ . Untuk membuat diskriminasi adalah

$$D_i = V^T (x_i - \mu) \quad (4.5)$$

Dengan:

$D_i$  = diskriminator untuk tiap citra telapak tangan.

$V^T$  = matriks transpose dari eigen vektor yang sudah direduksi dengan PCA

$x_i$  = citra telapak tangan yang sudah di bentuk vektor kolom

$\mu$  = rata-rata dari semua citra di training set.

Adapun langkah-langkah untuk melakukan proses penghitungan diskriminator adalah sebagai berikut:



1. Memilih panjang diskriminator yang akan dibuat antara 50, 100, 150, atau 200 [4].
2. Ambil data citra telapak tangan ke- $i$  yang akan dicari diskriminatornya
3. Gunakan persamaan (4.5) untuk menghitung diskriminator citra telapak tangan yang ke- $i$
4. Ulangi langkah 2 dan 3 sampai semua citra tangan terbentuk diskriminatornya dengan panjang tertentu
5. Ulangi langkah 2-5 dengan mengganti panjang diskriminatornya, sampai terbentuk 4 jenis diskriminator untuk 1 citra telapak tangan dengan panjang yang berbeda.

#### 4.1.7 Proses Penyimpanan Telapak Tangan

Proses penyimpanan telapak tangan merupakan proses penyimpanan data-data milik pengguna yang melakukan pendaftaran atau registrasi pada basis data. Data-data yang disimpan berupa data citra telapak tangan pengguna yang direpresentasikan dengan vektor kolom,  $\mu$  yang merupakan rata-rata dari data citra training set, 4 matriks *eigen* vektor yang sudah direduksi, diskriminator, dan identitas pengguna. Semua data tadi disimpan dalam sebuah *file text* dan disimpan didalam folder dengan nama khusus sesuai jenis datanya. Khusus untuk citra wajah yang digunakan untuk identitas pengguna, disimpan di sebuah folder khusus dan alamat penyimpanannya disimpan di sebuah *file text*.

#### 4.1.8 Proses Pencocokan Telapak Tangan

Proses pencocokan telapak tangan merupakan sebuah proses pengukuran kesamaan antara diskriminator query dengan diskriminator referensi. Diskriminator query diperoleh dari citra telapak tangan pengguna yang akan melakukan identifikasi. Sedangkan diskriminator referensi merupakan diskriminator pengguna yang sudah tersimpan di basis data. Hasil dari proses pencocokan telapak tangan berupa suatu skor yang menentukan apakah pengguna berhasil diidentifikasi dengan benar atau tidak.

Pengukuran kesamaan antara diskriminator query, misal dilambangkan dengan  $u$ , dan diskriminator referensi, misal dilambangkan dengan  $v$ , dihitung dengan persamaan jarak Euclidean ternormalisasi sebagai berikut:

$$\bar{d}(u, v) = \left( \sum_{i=1}^n (\bar{u}_i - \bar{v}_i)^2 \right)^{\frac{1}{2}} \quad (4.6)$$

dengan:

$$\bar{u}_i = \frac{u_i}{\|u\|} \quad (4.7)$$

dan

$$\bar{v}_i = \frac{v_i}{\|v\|} \quad (4.8)$$

$\|u\|$  dan  $\|v\|$  masing-masing disebut dengan norm dari  $u$  dan norm dari  $v$  untuk jarak euclidean dan diperoleh dari persamaan sebagai berikut:

$$\|u\| = \left( \sum_{i=1}^n (u_i)^2 \right)^{\frac{1}{2}} \quad (4.9)$$

dan

$$\|v\| = \left( \sum_{i=1}^n (v_i)^2 \right)^{\frac{1}{2}} \quad (4.10)$$

dengan:

$\bar{d}(u, v)$  = jarak Euclidian ternormalisasi

$u$  = diskriminator dari citra yang diuji

$v$  = diskriminator dari citra yang telah didaftarkan

$\|u\|$  = norm dari  $u$

$\|v\|$  = norm dari  $v$

$\bar{u}$  = diskriminator query ternormalisasi

$\bar{v}$  = diskriminator referensi ternormalisasi

$n$  = panjang diskriminator

Sedangkan untuk mendapatkan skor hasil pencocokan telapak tangan maka digunakan metode pengukuran kesamaan (*similarity measure*). Rentang dari skor berada antara 0 sampai 1 yang didapat dengan persamaan sebagai berikut:

$$skor = 1 - \frac{\bar{d}(u,v)}{2} \quad (4.11)$$

Skor adalah suatu nilai yang merepresentasikan tingkat kemiripan telapak tangan yang diuji dengan yang tersimpan di basis data. Semakin besar skornya semakin mirip citra yang dibandingkan.

Adapun langkah-langkah untuk melakukan proses pencocokan telapak tangan adalah sebagai berikut:

1. Akuisisi citra telapak tangan pengguna yang akan melakukan identifikasi. Citra telapak tangan ini digunakan sebagai citra telapak tangan input.
2. Membuat diskriminator untuk citra inputan yang akan diidentifikasi
3. Untuk setiap diskriminator referensi yang tersimpan di basis data hitung skor hasil pencocokan telapak tangan antara diskriminator query dan diskriminator referensi.
4. Diskriminator referensi yang menghasilkan skor terbesar pada langkah ke-3 dijadikan sebagai diskriminator teridentifikasi.
5. Jika skor dari diskriminator teridentifikasi lebih besar dari nilai ambang maka sistem berhasil mengidentifikasi pengguna.
6. Sebaliknya jika skor dari diskriminator teridentifikasi tidak lebih besar dari nilai ambang maka sistem tidak berhasil mengidentifikasi pengguna.

## **4.2 Implementasi Sistem**

### **4.2.1 Lingkungan Implementasi Sistem**

Lingkungan implementasi sistem yang dibangun meliputi beberapa perangkat keras dan perangkat lunak komputer. Detail spesifikasi lingkungan implementasi sistem dapat dilihat pada Tabel 4.3.

**Tabel 4.3 Detail Spesifikasi Lingkungan Implementasi Sistem**

|                 |                |                                                                                                  |
|-----------------|----------------|--------------------------------------------------------------------------------------------------|
| Perangkat Keras | Prosesor       | : Intel® Core™ 2 Duo CPU T6570 @2,10 GHz                                                         |
|                 | Memory         | : 4 GB DDR3                                                                                      |
| Perangkat Lunak | Sistem Operasi | : Windows 7 Ultimate                                                                             |
|                 | Tools          | : Microsoft Visual C# 2010<br>Aforge.NET ( <i>framework</i> )<br>Accord.NET ( <i>framework</i> ) |

#### **4.2.2 Implementasi Antarmuka Sistem**

Pada tugas akhir ini, antarmuka sistem dibangun dengan menggunakan form dan kontrol yang terdapat pada IDE Microsoft Visual C# 2010. IDE atau *Integrated Development Environment* Microsoft Visual C# 2010 merupakan sebuah tempat pembuatan program-program yang menggunakan bahasa pemrograman C#. Adapun antarmuka-antarmuka yang diimplementasikan untuk menunjang penelitian tugas akhir ini adalah sebagai berikut.

##### **4.2.2.1 Antarmuka Utama**

Antarmuka utama merupakan antarmuka yang berisi menu-menu untuk menampilkan antarmuka-antarmuka lainnya yang dimiliki oleh sistem. Hasil implementasi antarmuka utama dapat dilihat pada Gambar 4.4.



**Gambar 4.4: Antarmuka Utama**

Antar muka utama terdapat menu bar yang merupakan bagian yang berisi sederetan menu yang digunakan oleh sistem. Masing-masing menu memiliki beberapa submenu yang berfungsi untuk menjalankan berbagai perintah tertentu dalam sistem. Adapun menu-menu (beserta submenu yang dimiliki oleh setiap menu) yang ditampilkan pada antarmuka utama sistem disajikan dalam Tabel 4.4.

**Tabel 4.4 Kegunaan Menu dan Submenu Sistem**

| Menu   | Submenu       | Kegunaan                                                                                                |
|--------|---------------|---------------------------------------------------------------------------------------------------------|
| Proses | Registrasi    | Mendaftarkan identitas pengguna dan citra telapak tangan pengguna                                       |
|        | Identifikasi  | Melakukan identifikasi pengguna melalui <i>webcam</i> ataupun dengan membuka file yang sudah tersimpan. |
| Data   | Data Pengguna | Melihat data-data identitas yang sudah terdaftar di sistem.                                             |

|         |                           |                                                                                                        |
|---------|---------------------------|--------------------------------------------------------------------------------------------------------|
|         | Eigen Vektor<br>Tereduksi | Menampilkan data matriks<br>eigen vektor yang sudah<br>direduksi sesuai dengan<br>ukuran diskriminator |
| Bantuan | Petunjuk<br>Penggunaan    | Menampilkan petunjuk<br>penggunaan sistem                                                              |
|         | Tentang                   | Menampilkan keterangan<br>tentang sistem ini.                                                          |

#### 4.2.2.2 Antarmuka Registrasi Pengguna

Antarmuka registrasi pengguna digunakan untuk menginputkan identitas pengguna yang akan melakukan registrasi pada sistem dan mendaftarkan citra telapak tangannya. Hasil implementasi antarmuka registrasi pengguna dapat dilihat pada Gambar 4.5.

**Gambar 4.5: Antarmuka Registrasi Data Pengguna**

Pada antarmuka registrasi pengguna ini terdapat 6 buah tombol yang memiliki fungsi yang berbeda-beda. Fungsi dari tombol-tombol tersebut adalah :

1. Tombol Start (Atas) : digunakan untuk mengakuisisi / *mengcapture* citra wajah yang digunakan untuk identitas pengguna melalui *webcam*. Tombol ini akan membuka antarmuka pilihan kamera untuk memilih kamera/*webcam* mana yang akan dipilih.
2. Tombol File (Atas) : digunakan untuk membuka file citra wajah pengguna yang sudah tersimpan di komputer.
3. Tombol Start (Bawah) : digunakan untuk mengakuisisi / *mengcapture* citra telapak tangan yang nantinya akan diekstraksi cirinya langsung melalui *webcam*. Tombol ini akan membuka antarmuka pilihan kamera untuk memilih kamera/*webcam* mana yang akan dipilih.
4. Tombol File (Bawah) : digunakan untuk membuka file citra telapak tangan yang sudah tersimpan di komputer.
5. Tombol Simpan : digunakan untuk mendaftarkan identitas pengguna ke dalam sistem, dan melakukan proses ekstraksi ciri dari telapak tangan pengguna.
6. Tombol Batal : digunakan untuk membatalkan proses registrasi.

#### **4.2.2.3 Antarmuka Pilihan Kamera**

Antarmuka pilihan kamera berguna untuk menampilkan pilihan kamera / *webcam* mana yang akan digunakan untuk mengakuisisi citra wajah ataupun telapak tangan pengguna. Jumlah pilihan kamera yang dipilih sesuai dengan driver kamera yang sudah terinstal di komputer itu sendiri. Pada antarmuka ini terdapat 2 tombol, tombol refresh digunakan untuk merefresh pilihan-pilihan kamera. Tombol pilih digunakan untuk memilih kamera /*webcam* yang akan digunakan ketika proses akuisisi. Antarmuka pilihan kamera bisa dilihat pada Gambar 4.6.



**Gambar 4.6: Antarmuka Pilihan Kamera**

#### 4.2.2.4 Antarmuka Identifikasi Pengguna

Antarmuka hasil identifikasi pengguna digunakan untuk mengidentifikasi telapak tangan pengguna. Hasil implementasi antarmuka hasil identifikasi pengguna dapat dilihat pada Gambar 4.7.



**Gambar 4.7: Antarmuka Identifikasi Pengguna**

Pada antarmuka ini terdapat beberapa tombol yang dapat digunakan, meliputi: tombol Start untuk akuisisi citra telapak tangan melalui *webcam*, tombol File untuk membuka file citra telapak tangan yang sudah tersimpan di komputer, dan tombol Proses untuk memulai proses identifikasi pengguna. Selain itu terdapat sebuah *combo box* yang digunakan untuk mengubah-ubah panjang diskriminator sesuai dengan yang diinginkan, dan *link label* Hasil Skor Detail yang digunakan untuk menampilkan antarmuka hasil skor yang digunakan untuk melihat hasil



keseluruhan skor dari proses identifikasi pengguna. Jika hasil skor tertinggi tidak melewati nilai ambang (*threshold*) maka antarmuka identifikasi pengguna akan mengeluarkan identitas kosong.

#### 4.2.2.5 Antarmuka Hasil Skor

Antarmuka hasil skor seluruh pengguna digunakan untuk menampilkan hasil proses identifikasi telapak tangan terhadap seluruh pengguna yang sudah terdaftar pada sistem. Hasil implementasi dari antarmuka ini dapat dilihat pada Gambar 4.8.

Form Hasil Skor

| Skor Hasil Identifikasi |                         |                      |
|-------------------------|-------------------------|----------------------|
| No Identitas            | Nama                    | Hasil Skor           |
| 1210100033              | Dyah Ayu E              | 0.000906335690454473 |
| 1210100025              | Nyoman Genry S          | 0.000408831827500334 |
| 1210100014              | Siti Lukmatul           | 0.000609072346900656 |
| 1210100026              | Mohamad Muhtaromi       | 0.000602583566565664 |
| 1210100055              | Nurani                  | 0.000590894356926343 |
| 1213100021              | Fara                    | 0.000588001380869518 |
| 1210100047              | Meyrina Harawati        | 0.000580208014899864 |
| 1210100032              | Zendhestara             | 0.000596445490007463 |
| 1212100065              | Mochamad Ihami          | 0.000550953358407886 |
| 1210100029              | Amilia K.M              | 0.000538683330418722 |
| 1210100013              | Ayu Firdha Y            | 0.000535466155441401 |
| 1210100021              | Raina M                 | 0.000527978948079361 |
| 1210100056              | Firdha DZ               | 0.000527611628360935 |
| 123                     | Nuzul Hidayati          | 0.000526749747367706 |
| 1212100076              | Zulfar                  | 0.000526254282919135 |
| 1210100052              | Ahmad Sirajuddin Luthfi | 0.000524790914750921 |
| 1210100012              | Ahmad S Badar           | 0.000519306092486822 |
| 1210100703              | Lulu Fajar Ramadhan     | 0.000488996475781763 |
| 1210100049              | Akhmad Khurraeni        | 0.000458911538408347 |
| 1210100074              | Edi Krisnayana          | 0.000458099031223161 |

Gambar 4.8: Antarmuka Hasil Skor

#### 4.2.2.6 Antarmuka Data Pengguna

Antarmuka data pengguna berisi identitas-identitas pengguna yang sudah terdaftar pada sistem. Hasil implementasi antarmuka data pengguna dapat dilihat pada Gambar 4.9.

| No identitas | Nama           | Asal         |
|--------------|----------------|--------------|
| 1210100060   | M Romdoni A    | Sidoarjo     |
| 1213100021   | Fara           | Surabaya     |
| 1210100012   | Ahmad S Badari | Tulung Agung |
| 1213100031   | Zani           | Gresik       |
| 1210100055   | Nursani        | Pasuruan     |
| 1210100032   | Zendriastara   | Sidoarjo     |
| 1212100065   | Mochamad Iham  | Lumajang     |
| 1210100072   | Gusrotu Any J  | Blau         |
| 1210100014   | Sh Lukmabul    | Bojonegara   |
| 1210100011   | Zaky Nur H     | Sidoarjo     |
| 1210100025   | Nyoman Genny S | Lombok       |

**Gambar 4.9: Antarmuka Data Pengguna**

Pada antarmuka ini tombol hapus yang digunakan untuk menghapus data pengguna yang telah dipilih sebelumnya pada tabel data pengguna.

#### 4.2.2.7 Antarmuka Vektor Eigen Tereduksi

Antarmuka vektor eigen tereduksi digunakan untuk menampilkan informasi mengenai 4 jenis matriks vektor eigen yang sudah direduksi sesuai panjang diskriminator. Hasil

implementasi antarmuka vektor eigen tereduksi dapat dilihat pada Gambar 4.10.

The image shows a screenshot of a software application window titled "Vektor Eigen Tereeduksi". The window has a green header bar. Below the header, there is a table displaying a large matrix of numerical values. The table has multiple columns and rows, with the first column containing labels that appear to be indices or identifiers. The numerical values are small and densely packed, typical of a matrix representation in a software interface.

**Gambar 4.10: Antarmuka Vektor Eigen Tereeduksi**

Ukuran matriks yang ditampilkan adalah  $1024 \times N$  dengan  $N = 50, 100, 150$ , dan  $200$  [4]. Untuk mengganti ukuran matriks yang ditampilkan dapat menggunakan *combo box* panjang diskriminator.

#### 4.2.2.8 Antarmuka Pra-Pengolahan

Antarmuka Pra-Pengolahan menampilkan secara detail proses pra-pengolahan dimana ini adalah proses agar citra telapak tangan itu siap diekstraksi cirinya. Hasil antarmuka pra-pengolahan dapat dilihat pada Gambar 4.11.



**Gambar 4.11: Antarmuka Pra-Pengolahan**

#### **4.2.3 Implementasi Proses Akuisisi Citra Telapak Tangan**

Proses akuisisi citra telapak tangan digunakan untuk mendapatkan citra telapak tangan pengguna, baik dilakukan secara langsung maupun tidak langsung. Untuk mendapatkan citra telapak tangan secara langsung maka citra telapak tangan diakuisisi dengan menggunakan *webcam*. Sedangkan untuk mendapatkan citra telapak tangan secara tidak langsung maka citra telapak tangan diambil dari data citra yang sebelumnya telah tersimpan di hardisk komputer.

Pada proses akuisisi citra telapak tangan secara langsung, agar *webcam* dapat dideteksi oleh sistem maka *file* Aforge.Net harus diinstal. *File* Aforge.Net merupakan sebuah library yang berisi fungsi untuk pengembangan suatu perangkat lunak yang dapat digunakan oleh programmer untuk membaca hardware pada komputer dan untuk pemrosesan citra digital.

Proses akuisisi citra telapak tangan secara langsung diimplementasikan menjadi sebuah program ke dalam fungsi berikut ini:

```
private void video_NewFrame1(object sender,
NewFrameEventArgs eventArgs)
```

Sedangkan proses akuisisi citra telapak tangan secara tidak langsung diimplementasikan menjadi sebuah program ke dalam *event* tombol file :

```
private void btnFilePalmPrint_Click(object sender,
EventArgs e)
```

Kode program selengkapnya dari fungsi ini disajikan pada Lampiran A.

#### 4.2.4 Implementasi Proses Pra-pengolahan

Proses pra-pengolahan menggunakan fungsi yang sudah tersedia di dalam Aforge.NET, sehingga hanya perlu membuat obyek dari tipe data yang disediakan Aforge.NET. Proses ini di implementasikan dalam program menjadi 4 obyek:

```
Grayscale filterGrayscale;
ResizeBicubic resize;
ContrastStretch filterNormalization;
Sharpen filterSharp;
```

Untuk proses pengkonvesian menjadi vektor kolom, tidak menggunakan obyek dari Aforge.NET , implementasinya dalam sebuah fungsi :

```
public static double[] Konversi(Bitmap Citra)
```

Fungsi ini membutuhkan parameter citra yang akan dikonversinya, dan akan mengoutputkan sebuah vektor kolom yang merupakan representasi dari citra tadi. Kode program selengkapnya dari fungsi ini disajikan pada Lampiran B.

## 4.2.5 Implementasi Proses Ekstraksi Ciri

### 4.2.5.1 Implementasi Proses PCA

Proses PCA diimplementasikan menjadi sebuah program ke dalam fungsi berikut ini:

```
public static void PCA()
```

Fungsi ini membutuhkan fungsi-fungsi matematika dalam prosesnya, untuk menggunakannya maka harus diinstall *framework* Accord.NET. File Accord.NET dapat digunakan untuk pemrosesan yang bersifat matematis. Kode program selengkapnya dari fungsi ini disajikan pada Lampiran C.

### 4.2.5.2 Implementasi Proses Perhitungan Diskriminator

Proses penghitungan diskriminator diimplementasikan menjadi sebuah program ke dalam fungsi berikut ini:

```
public static double[] HitungDiskriminator(Bitmap  
Citra, int n)
```

Kode program selengkapnya dari fungsi ini disajikan pada Lampiran C. Fungsi ini menghasilkan sebuah array satu dimensi dengan tipe data double. Array ini merupakan hasil ekstraksi ciri citra telapak tangan. Parameter-parameter pada fungsi ini meliputi:

1. *Citra*, berupa array dua dimensi bertipe data double. Array ini berisi data citra telapak tangan.
2. *n* berupa bilangan dengan tipe data integer. Bilangan ini merepresentasikan panjang diskriminator yang akan dibuat.

Karena dalam tugas akhir ini ada 4 jenis diskriminator, maka fungsi ini dipanggil 4 kali.

## 4.2.6 Implementasi Proses Penyimpanan Telapak Tangan

Fungsi untuk menyimpan data citra telapak tangan pengguna terdapat pada fungsi berikut ini:

```
public void Simpan_PalmPrint(Bitmap Citra, string Id,
string Nama, string Asal, string AlamatFoto)
```

Dimana parameter dari fungsi ini digunakan untuk menyimpan data citra telapak tangan pengguna yang didaftarkan dan parameter lain digunakan untuk memanggil fungsi penyimpanan identitas pengguna. Sedangkan untuk fungsi penyimpanan diskriminator dari citra telapak tangan adalah :

```
public static void SimpanDiskriminator()
```

Fungsi ini berguna untuk menyimpan diskriminator dari suatu citra inputan menjadi 4 jenis diskriminator sesuai dengan panjang diskriminatornya. Kode program selengkapnya dari fungsi ini disajikan pada Lampiran D.

#### 4.2.7 Implementasi Proses Pencocokan Telapak Tangan

Proses pencocokan telapak tangan diimplementasikan menjadi sebuah program ke dalam fungsi berikut ini:

```
public static void IdentifikasiUser(Bitmap Citra , int
n)
```

Kode program selengkapnya dari fungsi ini disajikan pada Lampiran E. Fungsi ini digunakan untuk melakukan pencocokan telapak tangan pengguna yang akan diidentifikasi (pengguna query) dengan semua telapak tangan pengguna yang sebelumnya sudah tersimpan pada basis data sistem atau disebut dengan pengguna referensi. Pengguna dapat diidentifikasi berdasarkan skor yang dihasilkan antara diskriminator pengguna query dengan diskriminator pengguna referensi. Dari proses ini menghasilkan skor untuk setiap penggunanya, dari situ dicari skor terbesarnya. Jika skor terbesar melebihi nilai ambang maka pengguna teridentifikasi.

*“Halaman ini sengaja dikosongkan”*



## **BAB V**

### **HASIL DAN PEMBAHASAN PENGUJIAN SISTEM**

Bab ini menjelaskan mengenai proses pengujian yang dilakukan terhadap sistem identifikasi telapak tangan dengan menggunakan matriks diskriminator. Hasil pengujian kemudian dibahas untuk mengetahui unjuk kerja sistem secara keseluruhan dalam mengidentifikasi telapak tangan pengguna. Isi dari bab ini dimulai dengan mengenalkan lingkungan pengujian sistem yang digunakan. Bab ini membahas tentang pengujian proses pencocokan telapak tangan. Bab ini diakhiri dengan pembahasan hasil pengujian yang telah dilakukan.

#### **5.1 Lingkungan Pengujian Sistem**

Lingkungan pengujian sistem meliputi beberapa perangkat keras dan perangkat lunak komputer. Detail spesifikasi lingkungan pengujian sistem dapat dilihat pada Tabel 5.1.

**Tabel 5.1 Lingkungan Pengujian Sistem**

|                 |                |                                                                                             |
|-----------------|----------------|---------------------------------------------------------------------------------------------|
| Perangkat Keras | Prosesor       | : Intel® Core™ 2 Duo CPU T6570 @2,10 GHz                                                    |
|                 | Memory         | : 4 GB DDR3                                                                                 |
| Perangkat Lunak | Sistem Operasi | : Windows 7 Ultimate                                                                        |
|                 | Tools          | : Microsoft Visual C#<br>Aforge.NET ( <i>framework</i> )<br>Accord.NET ( <i>framework</i> ) |

#### **5.2 Pengujian Proses Pencocokan Telapak Tangan**

Pengujian proses pencocokan telapak tangan merupakan pengujian utama pada tugas akhir ini. Hal ini dikarenakan untuk mengetahui bagaimana unjuk kerja sistem identifikasi telapak tangan dengan menggunakan matriks diskriminator maka sistem harus diuji apakah sistem dapat digunakan untuk mengidentifikasi telapak tangan pengguna. Dalam hal identifikasi telapak tangan pengguna maka proses pencocokan telapak tangan memegang

peranan terpenting. Sebab pengguna berhasil atau gagal diidentifikasi berdasarkan skor yang dihasilkan dalam proses pencocokan telapak tangan.

Pada tugas akhir ini, pengujian proses pencocokan telapak tangan dilakukan dalam bentuk simulasi. Sistem diuji dengan pengguna yang sudah terdaftar pada sistem. Pengujian dengan pengguna yang sudah terdaftar untuk mengetahui tingkat kesalahan FNMR (*False Non Match Rate*). Kesalahan FNMR merupakan kesalahan dimana sistem tidak dapat mengidentifikasi pengguna yang sudah terdaftar pada sistem. Sistem ini akan diuji dengan perbedaan panjang diskriminator 50, 100, 150, dan 200.

Pada sistem, jumlah pengguna yang diuji adalah sebanyak 60 pengguna dibagi menjadi 2 sesi. Masing-masing pengguna terdaftar diakuisisi telapak tangannya sebanyak 3 kali. Telapak tangan yang diakuisisi adalah telapak tangan sebelah kiri.

### 5.2.1 Penentuan Nilai Ambang (*Threshold*)

Pada proses pencocokan telapak tangan, pengguna yang tersimpan pada basis data yang memiliki skor terbesar merupakan pengguna yang berhasil diidentifikasi oleh sistem. Agar dapat ditetapkan apakah pengguna tersebut sah atau tidak maka perlu ditentukan sebuah nilai ambang (*threshold*) sebagai batas pengambilan keputusan. Dengan demikian jika pengguna dengan skor terbesar memiliki skor yang lebih kecil dari *threshold* maka pengguna tersebut dianggap bukan pengguna yang teridentifikasi. Begitu juga sebaliknya, jika pengguna tersebut skornya lebih besar atau sama dengan *threshold* maka pengguna tersebut adalah pengguna yang teridentifikasi.

Penentuan *threshold* dilakukan berdasarkan pengamatan terhadap skor yang dihasilkan pada saat percobaan proses pencocokan telapak tangan, baik saat menggunakan diskriminator dengan panjang 50, 100, 150, ataupun 200 [4]. Dari hasil percobaan ditetapkan nilai *threshold* sebesar 0.001.

### **5.2.2 Pengujian Identifikasi Pengguna yang Terdaftar pada Sistem**

Pada pengujian ini setiap pengguna diakuisisi citranya sebanyak 3 kali. Citra telapak tangan kesatu untuk pendaftaran pengguna. Untuk 2 citra telapak tangan yang lain dan citra telapak tangan yang terdaftar digunakan sebagai pengujian. Citra telapak tangan kesatu didaftarkan dan diuji dengan 2 citra telapak tangan dan citra telapak tangan kesatu. Dalam tugas akhir akan didaftarkan 30 pengguna untuk sesi I dan 30 pengguna lain untuk sesi II. Setelah dilakukan pengujian pada sesi I pengguna yang terdaftar dihapus dan mendaftarkan 30 pengguna lain untuk pengujian sesi II. Jadi jumlah keseluruhan pengujian identifikasi pengguna yang terdaftar pada sistem adalah sebanyak  $3 \times 4$  jenis diskriminator  $\times$  jumlah pengguna yang terdaftar atau  $3 \times 4 \times 60 = 720$  pengujian.

Pengujian identifikasi pengguna yang terdaftar pada tiap sesi terdiri dari empat macam, yaitu pengujian identifikasi pengguna dengan menggunakan diskriminator dengan panjang 50, 100, 150, dan 200 [4].

Adapun sebagian hasil keempat macam pengujian identifikasi pengguna yang terdaftar pada sistem disajikan pada Tabel 5.2 sampai dengan Tabel 5.9.

**Tabel 5.2 Hasil Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi I dengan Menggunakan Diskriminator dengan panjang 50**

| No Identitas | Hasil Pengujian Citra |   |   |
|--------------|-----------------------|---|---|
|              | 1                     | 2 | 3 |
| 1209100068   | B                     | B | B |
| 1210100011   | B                     | G | G |
| 1210100012   | B                     | B | B |
| 1210100013   | B                     | B | B |
| 1210100019   | B                     | B | B |
| 1210100021   | B                     | B | B |
| 1210100025   | B                     | B | B |
| 1210100026   | B                     | B | B |
| 1210100027   | B                     | B | B |
| 1210100029   | B                     | B | B |
| 1210100032   | B                     | B | B |
| 1210100033   | B                     | B | B |
| 1210100047   | B                     | B | B |
| 1210100048   | B                     | B | B |
| 1210100049   | B                     | B | B |
| 1210100052   | B                     | B | B |
| 1210100055   | B                     | B | B |
| 1210100056   | B                     | B | B |
| 1210100060   | B                     | B | B |
| 1210100070   | B                     | B | B |
| 1210100072   | B                     | B | B |
| 1210100073   | B                     | B | B |
| 1210100074   | B                     | B | B |
| 1210100702   | B                     | B | B |
| 1210100703   | B                     | B | B |
| 1212100065   | B                     | B | B |
| 1212100076   | B                     | B | B |
| 1213100021   | B                     | G | B |
| 1213100031   | B                     | B | B |
| 1213100053   | B                     | G | B |

**Tabel 5.3 Hasil Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi I dengan Menggunakan Diskriminator dengan panjang 100**

| No Identitas | Hasil Pengujian Citra |   |   |
|--------------|-----------------------|---|---|
|              | 1                     | 2 | 3 |
| 1209100068   | B                     | B | B |
| 1210100011   | B                     | G | G |
| 1210100012   | B                     | B | B |
| 1210100013   | B                     | B | B |
| 1210100019   | B                     | B | B |
| 1210100021   | B                     | B | B |
| 1210100025   | B                     | B | B |
| 1210100026   | B                     | B | B |
| 1210100027   | B                     | B | B |
| 1210100029   | B                     | B | B |
| 1210100032   | B                     | B | B |
| 1210100033   | B                     | B | B |
| 1210100047   | B                     | B | B |
| 1210100048   | B                     | B | B |
| 1210100049   | B                     | B | B |
| 1210100052   | B                     | B | B |
| 1210100055   | B                     | B | B |
| 1210100056   | B                     | B | B |
| 1210100060   | B                     | B | B |
| 1210100070   | B                     | B | B |
| 1210100072   | B                     | B | B |
| 1210100073   | B                     | B | B |
| 1210100074   | B                     | B | B |
| 1210100702   | B                     | B | B |
| 1210100703   | B                     | B | B |
| 1212100065   | B                     | B | B |
| 1212100076   | B                     | B | B |
| 1213100021   | B                     | G | B |
| 1213100031   | B                     | B | B |
| 1213100053   | B                     | G | B |

**Tabel 5.4 Hasil Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi I dengan Menggunakan Diskriminator dengan panjang 150**

| No Identitas | Hasil Pengujian Citra |   |   |
|--------------|-----------------------|---|---|
|              | 1                     | 2 | 3 |
| 1209100068   | B                     | B | B |
| 1210100011   | B                     | G | G |
| 1210100012   | B                     | B | B |
| 1210100013   | B                     | B | B |
| 1210100019   | B                     | B | B |
| 1210100021   | B                     | B | B |
| 1210100025   | B                     | B | B |
| 1210100026   | B                     | B | B |
| 1210100027   | B                     | B | B |
| 1210100029   | B                     | B | B |
| 1210100032   | B                     | B | B |
| 1210100033   | B                     | B | B |
| 1210100047   | B                     | B | B |
| 1210100048   | B                     | B | B |
| 1210100049   | B                     | B | B |
| 1210100052   | B                     | B | B |
| 1210100055   | B                     | B | B |
| 1210100056   | B                     | B | B |
| 1210100060   | B                     | B | B |
| 1210100070   | B                     | B | B |
| 1210100072   | B                     | B | B |
| 1210100073   | B                     | B | B |
| 1210100074   | B                     | B | B |
| 1210100702   | B                     | B | B |
| 1210100703   | B                     | B | B |
| 1212100065   | B                     | B | B |
| 1212100076   | B                     | B | B |
| 1213100021   | B                     | G | B |
| 1213100031   | B                     | B | G |
| 1213100053   | B                     | G | B |

**Tabel 5.5 Hasil Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi I dengan Menggunakan Diskriminator dengan panjang 200**

| No Identitas | Hasil Pengujian Citra |   |   |
|--------------|-----------------------|---|---|
|              | 1                     | 2 | 3 |
| 1209100068   | B                     | B | B |
| 1210100011   | B                     | G | G |
| 1210100012   | B                     | B | B |
| 1210100013   | B                     | B | B |
| 1210100019   | B                     | B | B |
| 1210100021   | B                     | B | B |
| 1210100025   | B                     | B | B |
| 1210100026   | B                     | B | B |
| 1210100027   | B                     | B | B |
| 1210100029   | B                     | B | B |
| 1210100032   | B                     | B | B |
| 1210100033   | B                     | B | B |
| 1210100047   | B                     | B | B |
| 1210100048   | B                     | B | B |
| 1210100049   | B                     | B | B |
| 1210100052   | B                     | B | B |
| 1210100055   | B                     | B | B |
| 1210100056   | B                     | B | B |
| 1210100060   | B                     | B | B |
| 1210100070   | B                     | B | B |
| 1210100072   | B                     | B | B |
| 1210100073   | B                     | B | B |
| 1210100074   | B                     | B | B |
| 1210100702   | B                     | B | B |
| 1210100703   | B                     | B | B |
| 1212100065   | B                     | B | B |
| 1212100076   | B                     | B | B |
| 1213100021   | B                     | G | B |
| 1213100031   | B                     | B | G |
| 1213100053   | B                     | G | B |

**Tabel 5.6 Hasil Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi II dengan Menggunakan Diskriminator dengan panjang 50**

| No Identitas | Hasil Pengujian Citra |   |   |
|--------------|-----------------------|---|---|
|              | 1                     | 2 | 3 |
| 1208100050   | B                     | G | G |
| 1209100035   | B                     | B | B |
| 1209100085   | B                     | B | B |
| 1210100002   | B                     | B | B |
| 1210100010   | B                     | B | B |
| 1210100020   | B                     | B | B |
| 1210100023   | B                     | B | B |
| 1210100030   | B                     | G | B |
| 1210100036   | B                     | B | B |
| 1210100039   | B                     | B | B |
| 1210100042   | B                     | B | B |
| 1210100070   | B                     | B | B |
| 1211100006   | B                     | B | B |
| 1211100037   | B                     | B | B |
| 1211100050   | B                     | B | B |
| 1211100065   | B                     | B | B |
| 1211100066   | B                     | B | B |
| 1211100070   | B                     | B | B |
| 1211100074   | B                     | B | B |
| 1211100091   | B                     | B | B |
| 1211100106   | B                     | B | B |
| 1211100110   | B                     | B | B |
| 1211100113   | B                     | B | B |
| 1212100011   | B                     | B | B |
| 1212100026   | B                     | B | B |
| 1212100044   | B                     | B | B |
| 1212100053   | B                     | B | B |
| 1212100059   | B                     | B | B |
| 1212100078   | B                     | B | B |
| 1213100088   | B                     | B | G |



**Tabel 5.7 Hasil Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi II dengan Menggunakan Diskriminator dengan panjang 100**

| No Identitas | Hasil Pengujian Citra |   |   |
|--------------|-----------------------|---|---|
|              | 1                     | 2 | 3 |
| 1208100050   | B                     | G | G |
| 1209100035   | B                     | B | B |
| 1209100085   | B                     | B | B |
| 1210100002   | B                     | B | B |
| 1210100010   | B                     | B | B |
| 1210100020   | B                     | B | B |
| 1210100023   | B                     | B | B |
| 1210100030   | B                     | G | B |
| 1210100036   | B                     | B | B |
| 1210100039   | B                     | B | B |
| 1210100042   | B                     | B | B |
| 1210100070   | B                     | B | B |
| 1211100006   | B                     | B | B |
| 1211100037   | B                     | G | B |
| 1211100050   | B                     | B | B |
| 1211100065   | B                     | B | B |
| 1211100066   | B                     | B | B |
| 1211100070   | B                     | B | B |
| 1211100074   | B                     | B | B |
| 1211100091   | B                     | B | B |
| 1211100106   | B                     | B | B |
| 1211100110   | B                     | B | B |
| 1211100113   | B                     | B | B |
| 1212100011   | B                     | B | B |
| 1212100026   | B                     | B | B |
| 1212100044   | B                     | B | B |
| 1212100053   | B                     | B | B |
| 1212100059   | B                     | B | B |
| 1212100078   | B                     | B | B |
| 1213100088   | B                     | B | G |

**Tabel 5.8 Hasil Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi II dengan Menggunakan Diskriminator dengan panjang 150**

| No Identitas | Hasil Pengujian Citra |   |   |
|--------------|-----------------------|---|---|
|              | 1                     | 2 | 3 |
| 1208100050   | B                     | G | G |
| 1209100035   | B                     | B | B |
| 1209100085   | B                     | B | B |
| 1210100002   | B                     | B | B |
| 1210100010   | B                     | B | B |
| 1210100020   | B                     | B | B |
| 1210100023   | B                     | B | B |
| 1210100030   | B                     | G | B |
| 1210100036   | B                     | B | B |
| 1210100039   | B                     | B | G |
| 1210100042   | B                     | B | B |
| 1210100070   | B                     | B | B |
| 1211100006   | B                     | B | B |
| 1211100037   | B                     | B | G |
| 1211100050   | B                     | B | B |
| 1211100065   | B                     | B | B |
| 1211100066   | B                     | B | B |
| 1211100070   | B                     | B | B |
| 1211100074   | B                     | B | B |
| 1211100091   | B                     | B | B |
| 1211100106   | B                     | B | B |
| 1211100110   | B                     | B | B |
| 1211100113   | B                     | B | B |
| 1212100011   | B                     | B | B |
| 1212100026   | B                     | B | B |
| 1212100044   | B                     | B | B |
| 1212100053   | B                     | B | B |
| 1212100059   | B                     | B | B |
| 1212100078   | B                     | B | B |
| 1213100088   | B                     | B | G |

**Tabel 5.9 Hasil Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi II dengan Menggunakan Diskriminator dengan panjang 200**

| No Identitas | Hasil Pengujian Citra |   |   |
|--------------|-----------------------|---|---|
|              | 1                     | 2 | 3 |
| 1208100050   | B                     | G | G |
| 1209100035   | B                     | B | B |
| 1209100085   | B                     | B | B |
| 1210100002   | B                     | B | B |
| 1210100010   | B                     | B | B |
| 1210100020   | B                     | B | B |
| 1210100023   | B                     | B | B |
| 1210100030   | B                     | G | B |
| 1210100036   | B                     | B | B |
| 1210100039   | B                     | B | G |
| 1210100042   | B                     | B | B |
| 1210100070   | B                     | B | B |
| 1211100006   | B                     | B | B |
| 1211100037   | B                     | B | G |
| 1211100050   | B                     | B | B |
| 1211100065   | B                     | B | B |
| 1211100066   | B                     | B | G |
| 1211100070   | B                     | B | B |
| 1211100074   | B                     | B | B |
| 1211100091   | B                     | B | B |
| 1211100106   | B                     | B | B |
| 1211100110   | B                     | B | B |
| 1211100113   | B                     | B | B |
| 1212100011   | B                     | B | B |
| 1212100026   | B                     | B | B |
| 1212100044   | B                     | B | B |
| 1212100053   | B                     | B | B |
| 1212100059   | B                     | B | B |
| 1212100078   | B                     | B | B |
| 1213100088   | B                     | B | G |

Pada Tabel 5.2 sampai dengan Tabel 5.9, **B** menunjukkan bahwa sistem berhasil mengidentifikasi pengguna dengan benar. Sedangkan **G** menunjukkan bahwa sistem gagal mengidentifikasi pengguna dengan benar. Kegagalan sistem dalam mengidentifikasi, yaitu tidak ada identitas siapapun yang berhasil diidentifikasi oleh sistem atau dengan kata lain identitas yang diidentifikasi adalah kosong.

Berdasarkan hasil pengujian identifikasi pengguna yang terdaftar pada sistem diperoleh beberapa data hasil kinerja sistem identifikasi telapak tangan dengan menggunakan matriks diskriminator. Data-data ini ditampilkan pada Tabel 5.10 sampai dengan Tabel 5.17.

**Tabel 5.10 Hasil Kinerja Sistem Berdasarkan Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi I dengan Menggunakan Diskriminator dengan panjang 50**

| No | Kriteria                         | Hasil |
|----|----------------------------------|-------|
| 1  | Jumlah pengujian Identifikasi    | 90    |
| 2  | Jumlah keberhasilan Identifikasi | 86    |
| 3  | Jumlah kegagalan Identifikasi    | 4     |

**Tabel 5.11 Hasil Kinerja Sistem Berdasarkan Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi I dengan Menggunakan Diskriminator dengan panjang 100**

| No | Kriteria                         | Hasil |
|----|----------------------------------|-------|
| 1  | Jumlah pengujian Identifikasi    | 90    |
| 2  | Jumlah keberhasilan Identifikasi | 86    |
| 3  | Jumlah kegagalan Identifikasi    | 4     |

**Tabel 5.12 Hasil Kinerja Sistem Berdasarkan Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi I dengan Menggunakan Diskriminator dengan panjang 150**

| No | Kriteria                         | Hasil |
|----|----------------------------------|-------|
| 1  | Jumlah pengujian Identifikasi    | 90    |
| 2  | Jumlah keberhasilan Identifikasi | 85    |
| 3  | Jumlah kegagalan Identifikasi    | 5     |

**Tabel 5.13 Hasil Kinerja Sistem Berdasarkan Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi I dengan Menggunakan Diskriminator dengan panjang 200**

| No | Kriteria                         | Hasil |
|----|----------------------------------|-------|
| 1  | Jumlah pengujian Identifikasi    | 90    |
| 2  | Jumlah keberhasilan Identifikasi | 85    |
| 3  | Jumlah kegagalan Identifikasi    | 5     |

**Tabel 5.14 Hasil Kinerja Sistem Berdasarkan Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi II dengan Menggunakan Diskriminator dengan panjang 50**

| No | Kriteria                         | Hasil |
|----|----------------------------------|-------|
| 1  | Jumlah pengujian Identifikasi    | 90    |
| 2  | Jumlah keberhasilan Identifikasi | 86    |
| 3  | Jumlah kegagalan Identifikasi    | 4     |

**Tabel 5.15 Hasil Kinerja Sistem Berdasarkan Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi II dengan Menggunakan Diskriminator dengan panjang 100**

| No | Kriteria                         | Hasil |
|----|----------------------------------|-------|
| 1  | Jumlah pengujian Identifikasi    | 90    |
| 2  | Jumlah keberhasilan Identifikasi | 85    |
| 3  | Jumlah kegagalan Identifikasi    | 5     |

**Tabel 5.16 Hasil Kinerja Sistem Berdasarkan Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi II dengan Menggunakan Diskriminator dengan panjang 150**

| No | Kriteria                         | Hasil |
|----|----------------------------------|-------|
| 1  | Jumlah pengujian Identifikasi    | 90    |
| 2  | Jumlah keberhasilan Identifikasi | 84    |
| 3  | Jumlah kegagalan Identifikasi    | 6     |

**Tabel 5.17 Hasil Kinerja Sistem Berdasarkan Pengujian Identifikasi Pengguna yang Terdaftar pada Sesi II dengan Menggunakan Diskriminator dengan panjang 200**

| No | Kriteria                         | Hasil |
|----|----------------------------------|-------|
| 1  | Jumlah pengujian Identifikasi    | 90    |
| 2  | Jumlah keberhasilan Identifikasi | 83    |
| 3  | Jumlah kegagalan Identifikasi    | 7     |

### 5.3 Pembahasan Hasil Pengujian

Pembahasan hasil pengujian difokuskan pada hasil pengujian proses pencocokan telapak tangan yang menjadi pengujian utama pada tugas akhir ini. Pembahasan hasil pengujian ini digunakan untuk mengetahui hasil kinerja sistem identifikasi telapak tangan dengan menggunakan matriks diskriminator secara keseluruhan. Di samping itu, pembahasan hasil pengujian juga digunakan untuk menarik kesimpulan tentang penyelesaian permasalahan yang dihadapi pada tugas akhir.

#### 5.3.1 Pembahasan Hasil Pengujian Identifikasi Pengguna yang Terdaftar pada Sistem

Tujuan dari pengujian pengguna yang terdaftar pada sistem adalah untuk mengetahui tingkat kesalahan FNMR (*False Non Match Rate*) pada sistem. Tingkat kesalahan FNMR dihasilkan dari persentase perbandingan jumlah kegagalan identifikasi pengguna yang terdaftar pada sistem dengan jumlah

keseluruhan pengujian atau dapat dirumuskan sebagai berikut [31] :

$$FNMR = \frac{\text{jumlah kegagalan identifikasi}}{\text{jumlah pengujian}} \times 100\% \quad (5.1)$$

Berdasarkan Tabel 5.10 sampai Tabel 5.17 dapat dihitung tingkat kesalahan FNMR untuk setiap proses pengujian adalah ditunjukkan pada Tabel 5.18 sampai Tabel 5.20

**Tabel 5.18 Tingkat Kesalahan FNMR Hasil Pengujian Pengguna yang Terdaftar pada Sistem untuk Sesi I**

| No | Panjang Diskriminator | Tingkat Kesalahan FNMR |
|----|-----------------------|------------------------|
| 1  | 50                    | 4,44%                  |
| 2  | 100                   | 4,44%                  |
| 3  | 150                   | 5,55%                  |
| 4  | 200                   | 5,55%                  |

**Tabel 5.19 Tingkat Kesalahan FNMR Hasil Pengujian Pengguna yang Terdaftar pada Sistem untuk Sesi II**

| No | Panjang Diskriminator | Tingkat Kesalahan FNMR |
|----|-----------------------|------------------------|
| 1  | 50                    | 4,44%                  |
| 2  | 100                   | 5,55%                  |
| 3  | 150                   | 6,66%                  |
| 4  | 200                   | 7,77%                  |

**Tabel 5.20 Rata-Rata Tingkat Kesalahan FNMR Hasil Pengujian Pengguna yang Terdaftar pada Sistem**

| No | Panjang Diskriminator | Tingkat Kesalahan FNMR |
|----|-----------------------|------------------------|
| 1  | 50                    | 4,44%                  |
| 2  | 100                   | 4,99%                  |
| 3  | 150                   | 6,11%                  |
| 4  | 200                   | 6,66%                  |

Berdasarkan tabel 5.18 sampai tabel 5.20 dapat ditunjukkan bahwa penggunaan diskiriminotor dengan panjang 50

pada pengujian pengguna yang terdaftar pada sistem memberikan hasil yang lebih baik jika dibandingkan dengan penggunaan diskriminator dengan panjang 100, 150, ataupun 200. Hal ini dapat diketahui dari rata-rata tingkat kesalahan FNMR untuk kedua sesi. Dengan semakin kecilnya tingkat kesalahan FNMR maka persentase keberhasilan identifikasi sistem menjadi semakin besar. Persentase keberhasilan sistem dapat dihitung dengan menggunakan persamaan berikut [3] :

$$\text{Persentase keberhasilan} = 100\% - \text{FNMR} \quad (5.52)$$

Dengan menggunakan hasil pada tabel 5.18 sampai tabel 5.20 maka dapat dihitung persentase keberhasilan sistem untuk setiap proses pengujian yang ditunjukkan pada tabel 5.21 sampai tabel 5.23.

**Tabel 5.21 Persentase Keberhasilan Sistem Hasil Pengujian Pengguna yang Terdaftar pada Sistem untuk Sesi I**

| No | Panjang Diskriminator | Presentase Keberhasilan |
|----|-----------------------|-------------------------|
| 1  | 50                    | 95,56%                  |
| 2  | 100                   | 95,56%                  |
| 3  | 150                   | 94,45%                  |
| 4  | 200                   | 94,45%                  |

**Tabel 5.22 Persentase Keberhasilan Sistem Hasil Pengujian Pengguna yang Terdaftar pada Sistem untuk Sesi II**

| No | Panjang Diskriminator | Presentase Keberhasilan |
|----|-----------------------|-------------------------|
| 1  | 50                    | 95,56%                  |
| 2  | 100                   | 94,45%                  |
| 3  | 150                   | 93,34%                  |
| 4  | 200                   | 92,23%                  |



**Tabel 5.23 Rata-Rata Persentase Keberhasilan Sistem Hasil Pengujian Pengguna yang Terdaftar pada Sistem**

| <b>No</b> | <b>Panjang Diskriminator</b> | <b>Presentase Keberhasilan</b> |
|-----------|------------------------------|--------------------------------|
| 1         | 50                           | 95,56%                         |
| 2         | 100                          | 95,01%                         |
| 3         | 150                          | 93,90%                         |
| 4         | 200                          | 93.34%                         |

### **5.3.2 Rekapitulasi Hasil Pengujian Secara Keseluruhan**

Berdasarkan pembahasan hasil pengujian pada subbab 5.3.1 dapat dilihat bahwa sistem identifikasi telapak tangan menggunakan matriks diskriminator memiliki tingkat kesalahan FNMR minimal pada diskriminator dengan panjang 50, yaitu sebesar 4,44% dan memiliki tingkat keberhasilan maksimal yaitu 95,56%.

*“Halaman ini sengaja dikosongkan”*

## **BAB VI PENUTUP**

Bab ini berisi tentang beberapa kesimpulan yang dihasilkan berdasarkan penelitian yang telah dilaksanakan. Di samping itu, pada bab ini juga dimasukkan beberapa saran yang dapat digunakan jika penelitian ini ingin dikembangkan.

### **6.1 Kesimpulan**

Berdasarkan analisis terhadap hasil pengujian sistem identifikasi telapak tangan dengan matriks diskriminator, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Dalam tugas akhir ini diskriminator yang dihasilkan dari PCA digunakan untuk melakukan identifikasi telapak tangan. Proses pencocokan telapak tangan dihitung dengan menggunakan persamaan jarak Euclidean ternormalisasi dan penghitungan skor menggunakan *similarity measure*.
2. Diskriminator yang dihasilkan dari PCA memiliki hasil yang bagus dalam mengekstraksi dan merepresentasikan ciri citra telapak tangan. Hal ini dapat diketahui berdasarkan persentase tingkat keberhasilan sistem dalam mengidentifikasi pengguna yang dapat mencapai 95,56% pada pengujian pengguna dengan panjang diskriminator 50.

### **6.2 Saran**

Dengan melihat hasil yang dicapai pada penelitian ini, ada beberapa hal yang penulis sarankan untuk pengembangan selanjutnya yaitu:

1. Sebelum citra telapak tangan diekstraksi cirinya, perlu adanya proses perbaikan citra telapak tangan untuk menutupi kemungkinan citra telapak tangan yang diakuisisi tidak begitu baik kualitasnya.
2. Sebaiknya digunakan suatu proses pra-pengolahan yang dapat mengatasi permasalahan pergeseran ataupun perputaran ketika proses akuisisi telapak tangan.

*“Halaman ini sengaja dikosongkan”*

## LAMPIRAN A

### A.1 Kode Prosedur Proses Akuisisi Citra Telapak Tangan Secara Langsung

```
private void video_NewFrame1(object sender,
NewFrameEventArgs eventArgs){
    Bitmap img = (Bitmap)eventArgs.Frame.Clone();
    //do processing here
    PbTelapakTangan.Image = img;
}
```

### A.2 Kode Untuk Mengakuisisi Citra dari File

```
private void btnFilePalmPrint_Click(object sender,
EventArgs e){
    OpenFileDialog openFileDialog = new OpenFileDialog();

    openFileDialog.ShowDialog(this);

    string imageFile = openFileDialog.FileName;

    if (imageFile == string.Empty) return;

    PbTelapakTangan.Image = Image.FromFile(imageFile);
}
```

## **LAMPIRAN A (LANJUTAN)**

*“Halaman ini sengaja dikosongkan”*

## LAMPIRAN B

### Proses Pra-Pengolahan

```
//GrayScale
Grayscale filterGrayscale = new Grayscale(0.2125,
0.7154, 0.0721);
Citra = filterGrayscale.Apply(Citra);

//Resize
ResizeBicubic resize = new ResizeBicubic(32, 32);
Citra = resize.Apply(Citra);

//ContrastStretch
ContrastStretch filterNormalization = new
ContrastStretch();
Citra = filterNormalization.Apply(Citra);

// create filter
Sharpen filterSharp = new Sharpen();
Citra = filterSharp.Apply(Citra);

public static double[] Konversi(Bitmap Citra)
{
    Citra = PraPengolahan(Citra);

    int x = Citra.Width, y = Citra.Height, c = 0;
    double[] Array_Citra = new double[x * y];

    for (int i = 0; i < x; i++)
    {
        for (int j = 0; j < y; j++)
        {
            Array_Citra[c] = Citra.GetPixel(i, j).R;
            c++;
        }
    }

    return Array_Citra;}

```

## **LAMPIRAN B (LANJUTAN)**

*“Halaman ini sengaja dikosongkan”*



## LAMPIRAN C

### C.1 Kode Fungsi Proses PCA

```

public static void PCA()
{
    //Hitung Jumlah Data
    Global.JmlPengguna =
    File.ReadAllLines("DataUser.txt").Length;

    //dapatkan data
    data = Data_Full();

    //data-mean
    _Mean = Mean(data);
    double[,] data_adjust = new double[D, JmlPengguna];

    for (int i = 0; i < D; i++)
    {
        for (int j = 0; j < JmlPengguna; j++)
            data_adjust[i, j] = data[i, j] - _Mean[i];
    }

    //cari Matriks Covarian
    double[,] cov =
    data_adjust.Multiply(data_adjust.Transpose());

    for (int i = 0; i < D; i++)
    {
        for (int j = 0; j < D; j++)
            cov[i, j] = cov[i, j] / JmlPengguna;
    }

    //mencari eigen value n eigen vektor
    var evd = new
    Accord.Math.Decompositions.EigenvalueDecomposition(cov)
    ;

    double[] eigenvalues = evd.RealEigenvalues;

```

## LAMPIRAN C (LANJUTAN)

```
eigenvectors = evd.Eigenvectors;

//sorting eigenvektor berdasar eigen value;
eigenvectors = Matrix.Sort(eigenvalues, eigenvectors,
new GeneralComparer(ComparerDirection.Descending,
true));

        SimpanDiskriminator();
    }
}
```

### C.2 Kode Fungsi Proses Penghitungan Diskriminator

```
public static double[] HitungDiskriminator(Bitmap
Citra, int n)
{
    double[] CitraQuery = Konversi(Citra);

    double[,] eigenvektor = DapatkanEigenVektor(n);
    double[] mean = DapatkanMean();

    double[] FiturQuery =
        eigenvektor.TransposeAndMultiply(CitraQuery.Subtract(mean));

    return FiturQuery;
}
```

## LAMPIRAN D

### D.1 Kode Fungsi Proses Penyimpanan Data Telapak Tangan

```
public void Simpan_PalmPrint(Bitmap Citra, string Id,
string Nama, string Asal, string AlamatFoto)
{
    double[] CitraKonversi = Global.Konversi(Citra);

    using (TextWriter tw = new
StreamWriter("DataPalmPrint\\" + Id + ".txt"))
    {
        for (int i = 0; i < CitraKonversi.Length; i++)
        {
            tw.WriteLine(CitraKonversi[i]);
        }
    }

    string str = Id + ";" + Nama + ";" + Asal + ";" +
AlamatFoto;

    SaveDataUser(str);
}
```

### D.2 Kode Fungsi Proses Penyimpanan Diskriminator

```
public static void SimpanDiskriminator()
{
    for (int i = 0; i < 4; i++)
    {
        double[,] HasilReduksi = new double[D,
PanjangDiskriminator[
i]];

        for (int j = 0; j < D; j++)
            for (int k = 0; k < PanjangDiskriminator[i];
k++)
                HasilReduksi[j, k] = eigenvectors[j, k];
    }
}
```

## LAMPIRAN D (LANJUTAN)

SimpanEigenVektorTereduksi(HasilReduksi,PanjangDiskriminator[i]);

```

    for (int l = 0; l < JmlPengguna; l++)
    {
        double[] Diskriminator = new double[D];

        for (int j = 0; j < D; j++)
        {
            Diskriminator[j] = data[j, l];
            Diskriminator[j] = Diskriminator[j] -
                _Mean[j];
        }

        using (TextWriter tw = new StreamWriter("FiturTangan\\"
+ PanjangDiskriminator[i] + "_Fitur\\" + Id[l] +
".txt"))
        {
            for (int j = 0; j < PanjangDiskriminator[i]; j++)
                tw.WriteLine(HasilReduksi.TransposeAndMultiply(Diskriminator)[j]);
        }
    }
}

```

## LAMPIRAN E

### Kode Fungsi Proses Identifikasi Pengguna

```
public static void IdentifikasiUser(Bitmap Citra , int
n)
{
    double[] Posisi = new double[JmlPengguna];
    double[] DiskriminatorQuery =
    HitungDiskriminator(Citra,n) ;
    double[] DiskriminatorReferensi = new double[n];
    HasilSkor = new double[JmlPengguna];
    EuclideanSimilarity Euclid = new
    EuclideanSimilarity();

    for(int j = 0; j<JmlPengguna; j++){
        string[] tmp = File.ReadAllLines("FiturTangan\\" +
        n + "_Fitur\\" + Id[j] + ".txt");

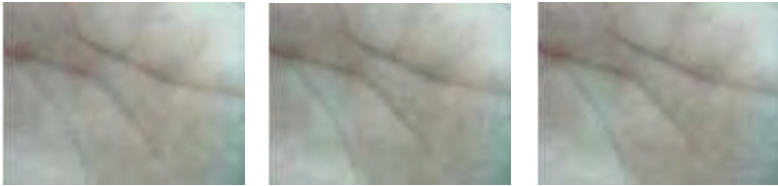
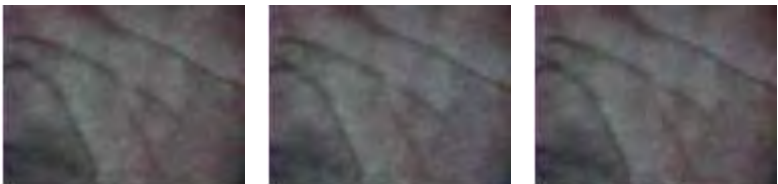
        for (int i = 0; i < tmp.Length; i++)
        {
            DiskriminatorReferensi[i] = double.Parse(tmp[i]);
        }

        HasilSkor[j]=
        Euclid.GetSimilarityScore(DiskriminatorReferensi,D
        iskriminatorQuery);
    }

    getMaxScore();
    DataSkoring();
}
```

**LAMPIRAN E (LANJUTAN)**

*“Halaman ini sengaja dikosongkan”*

**LAMPIRAN F****Sampel Data Citra Telapak Tangan untuk Pengujian****1211100106****1210100042****1209100035**

**LAMPIRAN F (LANJUTAN)**

*“Halaman ini sengaja dikosongkan”*



## BIODATA PENULIS



Penulis bernama lengkap Ahmad Sirojuddin Luthfi, biasa dipanggil Luthfi. Penulis dilahirkan di Surakarta pada tanggal 29 Juli 1992. Penulis lulus dari SDN Muhammadiyah 2 Surakarta, SMPN 3 Surakarta, SMAN 1 Surakarta dan melanjutkan pendidikan di Matematika ITS pada tahun 2010 melalui jalur SNMPTN.

Sejak kuliah penulis hobby mempelajari berbagai bahasa pemrograman. Bahasa pemrograman yang pernah penulis pelajari seperti C, C++, Java, Visual Basic.NET, dan C#. Oleh sebab itu, di jurusan Matematika penulis memilih bidang minat Ilmu Komputer.

Semasa menempuh jenjang pendidikan S-1, penulis juga aktif dalam kegiatan non-akademis di antaranya aktif di organisasi kegiatan kemahasiswaan Matematika ITS. Untuk mendapatkan informasi yang berhubungan dengan Tugas Akhir ini dapat ditunjukan ke alamat email: [ahmads.luthfi@gmail.com](mailto:ahmads.luthfi@gmail.com)